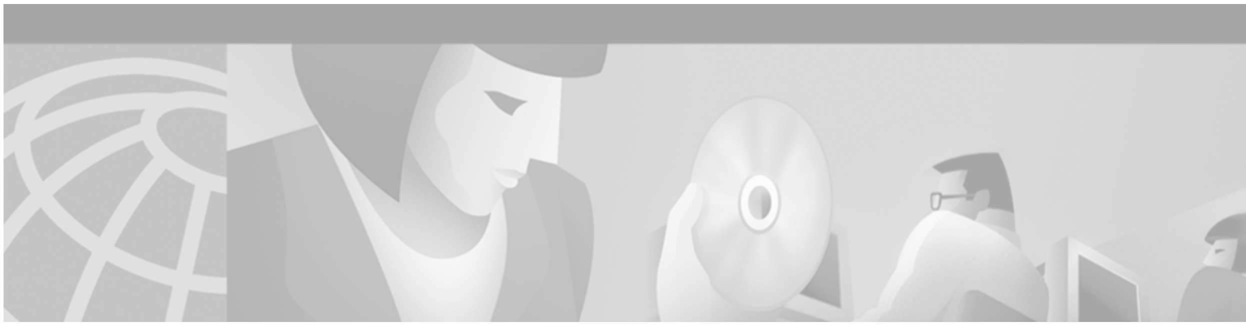

Configuring Cisco IOS Firewall Intrusion Detection System

This chapter describes the Cisco IOS Firewall Intrusion Detection System (IDS) feature. Intrusion detection systems provide a level of protection beyond the firewall by protecting the network from internal and external attacks and threats. Cisco IOS Firewall IDS technology enhances perimeter firewall protection by taking appropriate action on packets and flows that violate the security policy or represent malicious network activity.

For a complete description of the Cisco IOS Firewall IDS commands in this chapter, refer to the "Cisco IOS Firewall IDS Commands" chapter of the *Cisco IOS Security Command Reference*. To locate documentation of other commands that appear in this chapter, use the command reference master index or search online.

To identify the hardware platform or software image information associated with a feature, use the Feature Navigator on Cisco.com to search for information about the feature or refer to the software release notes for a specific release. For more information, see the chapter "Identifying Supported Platforms" section in the "Using Cisco IOS Software."

In This Chapter

This chapter has the following sections:

About the Firewall Intrusion Detection System

The Cisco IOS Firewall IDS feature supports intrusion detection technology for midrange and high-end router platforms with firewall support. It is ideal for any network perimeter, and especially for locations in which a router is being deployed and additional security between network segments is required. It also can protect intranet and extranet connections where additional security is mandated, and branch-office sites connecting to the corporate office or Internet.

The Cisco IOS Firewall IDS feature identifies 59 of the most common attacks using "signatures" to detect patterns of misuse in network traffic. The intrusion-detection signatures included in the Cisco IOS Firewall were chosen from a broad cross-section of intrusion-detection signatures. The signatures represent severe breaches of security and the most common network attacks and information-gathering scans. For a description of Cisco IOS Firewall IDS signatures, refer to the "Cisco IOS Firewall IDS Signature List" section.

The Cisco IOS Firewall IDS acts as an in-line intrusion detection sensor, watching packets and sessions as they flow through the router, scanning each to match any of the IDS signatures. IDS monitors packets and send alarms when suspicious activity is detected. IDS logs the event through Cisco IOS syslog or the Cisco Secure Intrusion Detection System (Cisco Secure IDS, formerly known as NetRanger) Post Office Protocol. The network administrator can configure the IDS system to choose the appropriate response to various threats. When packets in a session match a signature, the IDS system can be configured to take these actions:

- Send an alarm to a syslog server or a Cisco Secure IDS Director (centralized management interface)
- Drop the packet
- Reset the TCP connection

Cisco developed its Cisco IOS software-based intrusion-detection capabilities in Cisco IOS Firewall with flexibility in mind, so that individual signatures could be disabled in case of false positives. Also, while it is preferable to enable both the firewall and intrusion detection features of the CBAC security engine to support a network security policy, each of these features may be enabled independently and on different router interfaces. Cisco IOS software-based intrusion detection is part of the Cisco IOS Firewall.

This section has the following sections:

- Interaction with Cisco IOS Firewall Default Parameters
- Compatibility with Cisco Secure Intrusion Detection
- Functional Description
- When to Use Cisco IOS Firewall IDS
- Memory and Performance Impact
- Cisco IOS Firewall IDS Signature List

Interaction with Cisco IOS Firewall Default Parameters

When Cisco IOS IDS is enabled, Cisco IOS Firewall is automatically enabled. Thus, IDS uses Cisco IOS Firewall default parameter values to inspect incoming sessions. Default parameter values include the following:

- The rate at which IDS starts deleting half-open sessions (modified via the **ip inspect one-minute high** command)
- The rate at which IDS stops deleting half-open sessions (modified via the **ip inspect one-minute low** command)
- The maximum incomplete sessions (modified via the **ip inspect max-incomplete high** and the **ip inspect max-incomplete low** commands)

After the incoming TCP session setup rate crosses the one-minute high water mark, the router will reset the oldest half-open session, which is the default behavior of the Cisco IOS Firewall. Cisco IOS IDS cannot modify this default behavior. Thus, after a new TCP session rate crosses the one-minute high water mark and a router attempts to open new connections by sending SYN packets at the same time, the latest SYN packet will cause the router to reset the half-open session that was opened by the earlier SYN packet. Only the last SYN request will survive.

Compatibility with Cisco Secure Intrusion Detection

Cisco IOS Firewall is compatible with the Cisco Secure Intrusion Detection System (formally known as NetRanger). The Cisco Secure IDS is an enterprise-scale, real-time, intrusion detection system designed to detect, report, and terminate unauthorized activity throughout a network.

The Cisco Secure IDS consists of three components:

- Sensor
- Director
- Post Office

Cisco Secure IDS Sensors, which are high-speed network appliances, analyze the content and context of individual packets to determine if traffic is authorized. If a network's data stream exhibits unauthorized or suspicious activity, such as a SATAN attack, a ping sweep, or the transmission of a secret research project code word, Cisco Secure IDS Sensors can detect the policy violation in real time, forward alarms to a Cisco Secure IDS Director management console, and remove the offender from the network.

The Cisco Secure IDS Director is a high-performance, software-based management system that centrally monitors the activity of multiple Cisco Secure IDS Sensors located on local or remote network segments.

The Cisco Secure IDS Post Office is the communication backbone that allows Cisco Secure IDS services and hosts to communicate with each other. All communication is supported by a proprietary, connection-based protocol that can switch between alternate routes to maintain point-to-point connections.

Cisco Secure IDS customers can deploy the Cisco IOS Firewall IDS signatures to complement their existing IDS systems. This allows an IDS to be deployed to areas that may not be capable of supporting a Cisco Secure IDS Sensor. Cisco IOS Firewall IDS signatures can be deployed alongside or independently of other Cisco IOS Firewall features.

The Cisco IOS Firewall IDS can be added to the Cisco Secure IDS Director screen as an icon to provide a consistent view of all intrusion detection sensors throughout a network. The Cisco IOS Firewall intrusion detection capabilities have an enhanced reporting mechanism that permits logging to the Cisco Secure IDS Director console in addition to Cisco IOS syslog.

For additional information about Cisco Secure IDS (NetRanger), refer to the *NetRanger User Guide*.

Functional Description

The Cisco IOS Firewall IDS acts as an in-line intrusion detection sensor, watching packets as they traverse the router’s interfaces and acting upon them in a definable fashion. When a packet, or a number of packets in a session, match a signature, the Cisco IOS Firewall IDS may perform the following configurable actions:

- Alarm—Sends an alarm to a syslog server or Cisco Secure IDS Director
- Drop—Drops the packet
- Reset—Resets the TCP connection

The following describes the packet auditing process with Cisco IOS Firewall IDS:

- You create an audit rule, which specifies the signatures that should be applied to packet traffic and the actions to take when a match is found. An audit rule can apply informational and attack signatures to network packets. The signature list can have just one signature, all signatures, or any number of signatures in between. Signatures can be disabled in case of false positives or the needs of the network environment.
- You apply the audit rule to an interface on the router, specifying a traffic direction (*in* or *out*).
- If the audit rule is applied to the *in* direction of the interface, packets passing through the interface are audited before the inbound ACL has a chance to discard them. This allows an administrator to be alerted if an attack or information-gathering activity is underway even if the router would normally reject the activity.
- If the audit rule is applied to the *out* direction on the interface, packets are audited after they enter the router through another interface. In this case, the inbound ACL of the other interface may discard packets before they are audited. This may result in the loss of Cisco IOS Firewall IDS alarms even though the attack or information-gathering activity was thwarted.
- Packets going through the interface that match the audit rule are audited by a series of modules, starting with IP; then either ICMP, TCP, or UDP (as appropriate); and finally, the Application level.
- If a signature match is found in a module, then the following user-configured action(s) occur:
  - If the action is **alarm**, then the module completes its audit, sends an alarm, and passes the packet to the next module.
  - If the action is **drop**, then the packet is dropped from the module, discarded, and not sent to the next module.
  - If the action is **reset**, then the packets are forwarded to the next module, and packets with the reset flag set are sent to both participants of the session, if the session is TCP.

Note It is recommended that you use the **drop** and **reset** actions together.

If there are multiple signature matches in a module, only the first match fires an action. Additional matches in other modules fire additional alarms, but only one per module.

Note This process is different than on the Cisco Secure IDS Sensor appliance, which identifies all signature matches for each packet.

When to Use Cisco IOS Firewall IDS

Cisco IOS Firewall IDS capabilities are ideal for providing additional visibility at intranet, extranet, and branch-office Internet perimeters. Network administrators enjoy more robust protection against attacks on the network and can automatically respond to threats from internal or external hosts.

The Cisco IOS Firewall with intrusion detection is intended to satisfy the security goals of all of our customers, and is particularly appropriate for the following scenarios:

- Enterprise customers that are interested in a cost-effective method of extending their perimeter security across all network boundaries, specifically branch-office, intranet, and extranet perimeters.
- Small and medium-sized businesses that are looking for a cost-effective router that has an integrated firewall with intrusion-detection capabilities.

- Service provider customers that want to set up managed services, providing firewalling and intrusion detection to their customers, all housed within the necessary function of a router.

Memory and Performance Impact

The performance impact of intrusion detection will depend on the configuration of the signatures, the level of traffic on the router, the router platform, and other individual features enabled on the router such as encryption, source route bridging, and so on. Enabling or disabling individual signatures will not alter performance significantly, however, signatures that are configured to use Access Control Lists will have a significant performance impact.

Because this router is being used as a security device, no packet will be allowed to bypass the security mechanisms. The IDS process in the Cisco IOS Firewall router sits directly in the packet path and thus will search each packet for signature matches. In some cases, the entire packet will need to be searched, and state information and even application state and awareness must be maintained by the router.

For auditing atomic signatures, there is no traffic-dependent memory requirement. For auditing compound signatures, CBAC allocates memory to maintain the state of each session for each connection. Memory is also allocated for the configuration database and for internal caching.

Cisco IOS Firewall IDS Signature List

The following is a complete list of Cisco IOS Firewall IDS signatures. A signature detects patterns of misuse in network traffic. In Cisco IOS Firewall IDS, signatures are categorized into four types:

- Info Atomic
- Info Compound
- Attack Atomic
- Attack Compound

An info signature detects information-gathering activity, such as a port sweep.

An attack signature detects attacks attempted into the protected network, such as denial-of-service attempts or the execution of illegal commands during an FTP session.

Info and attack signatures can be either atomic or compound signatures. Atomic signatures can detect patterns as simple as an attempt to access a specific port on a specific host. Compound signatures can detect complex patterns, such as a sequence of operations distributed across multiple hosts over an arbitrary period of time.

The intrusion-detection signatures included in the Cisco IOS Firewall were chosen from a broad cross-section of intrusion-detection signatures as representative of the most common network attacks and information-gathering scans that are not commonly found in an operational network.

The following signatures are listed in numerical order by their signature number in the Cisco Secure IDS Network Security Database. After each signature's name is an indication of the type of signature (info or attack, atomic or compound).

Note Atomic signatures marked with an asterisk (Atomic*) are allocated memory for session states by CBAC.

1000 IP options-Bad Option List (Info, Atomic)

Triggers on receipt of an IP datagram where the list of IP options in the IP datagram header is incomplete or malformed. The IP options list contains one or more options that perform various network management or debugging tasks.

1001 IP options-Record Packet Route (Info, Atomic)

Triggers on receipt of an IP datagram where the IP option list for the datagram includes option 7 (Record Packet Route).

1002 IP options-Timestamp (Info, Atomic)

Triggers on receipt of an IP datagram where the IP option list for the datagram includes option 4 (Timestamp).

1003 IP options-Provide s,c,h,tcc (Info, Atomic)

Triggers on receipt of an IP datagram where the IP option list for the datagram includes option 2 (Security options).

1004 IP options-Loose Source Route (Info, Atomic)

Triggers on receipt of an IP datagram where the IP option list for the datagram includes option 3 (Loose Source Route).

1005 IP options-SATNET ID (Info, Atomic)

Triggers on receipt of an IP datagram where the IP option list for the datagram includes option 8 (SATNET stream identifier).

1006 IP options-Strict Source Route (Info, Atomic)

Triggers on receipt of an IP datagram in which the IP option list for the datagram includes option 2 (Strict Source Routing).

1100 IP Fragment Attack (Attack, Atomic)

Triggers when any IP datagram is received with the “more fragments” flag set to 1 or if there is an offset indicated in the offset field.

1101 Unknown IP Protocol (Attack, Atomic)

Triggers when an IP datagram is received with the protocol field set to 101 or greater. These protocol types are undefined or reserved and should not be used.

1102 Impossible IP Packet (Attack, Atomic)

This triggers when an IP packet arrives with source equal to destination address. This signature will catch the so-called Land Attack.

2000 ICMP Echo Reply (Info, Atomic)

Triggers when a IP datagram is received with the “protocol” field in the IP header set to 1 (ICMP) and the “type” field in the ICMP header set to 0 (Echo Reply).

2001 ICMP Host Unreachable (Info, Atomic)

Triggers when an IP datagram is received with the “protocol” field in the IP header set to 1 (ICMP) and the “type” field in the ICMP header set to 3 (Host Unreachable).

2002 ICMP Source Quench (Info, Atomic)

Triggers when an IP datagram is received with the “protocol” field in the IP header set to 1 (ICMP) and the “type” field in the ICMP header set to 4 (Source Quench).

2003 ICMP Redirect (Info, Atomic)

Triggers when an IP datagram is received with the “protocol” field in the IP header set to 1 (ICMP) and the “type” field in the ICMP header set to 5 (Redirect).

2004 ICMP Echo Request (Info, Atomic)

Triggers when an IP datagram is received with the “protocol” field in the IP header set to 1 (ICMP) and the “type” field in the ICMP header set to 8 (Echo Request).

2005 ICMP Time Exceeded for a Datagram (Info, Atomic)

Triggers when an IP datagram is received with the “protocol” field in the IP header set to 1 (ICMP) and the “type” field in the ICMP header set to 11(Time Exceeded for a Datagram).

2006 ICMP Parameter Problem on Datagram (Info, Atomic)

Triggers when an IP datagram is received with the “protocol” field in the IP header set to 1 (ICMP) and the “type” field in the ICMP header set to 12 (Parameter Problem on Datagram).

2007 ICMP Timestamp Request (Info, Atomic)

Triggers when an IP datagram is received with the “protocol” field in the IP header set to 1 (ICMP) and the “type” field in the ICMP header set to 13 (Timestamp Request).

2008 ICMP Timestamp Reply (Info, Atomic)

Triggers when an IP datagram is received with the “protocol” field in the IP header set to 1 (ICMP) and the “type” field in the ICMP header set to 14 (Timestamp Reply).

2009 ICMP Information Request (Info, Atomic)

Triggers when an IP datagram is received with the “protocol” field in the IP header set to 1 (ICMP) and the “type” field in the ICMP header set to 15 (Information Request).

2010 ICMP Information Reply (Info, Atomic)

Triggers when an IP datagram is received with the “protocol” field in the IP header set to 1 (ICMP) and the “type” field in the ICMP header set to 16 (ICMP Information Reply).

2011 ICMP Address Mask Request (Info, Atomic)

Triggers when an IP datagram is received with the “protocol” field in the IP header set to 1 (ICMP) and the “type” field in the ICMP header set to 17 (Address Mask Request).

2012 ICMP Address Mask Reply (Info, Atomic)

Triggers when an IP datagram is received with the “protocol” field in the IP header set to 1 (ICMP) and the “type” field in the ICMP header set to 18 (Address Mask Reply).

2150 Fragmented ICMP Traffic (Attack, Atomic)

Triggers when an IP datagram is received with the protocol field in the IP header set to 1 (ICMP) and either the more fragments flag is set to 1 (ICMP) or there is an offset indicated in the offset field.

2151 Large ICMP Traffic (Attack, Atomic)

Triggers when an IP datagram is received with the protocol field in the IP header set to 1 (ICMP) and the IP length is greater than 1024.

2154 Ping of Death Attack (Attack, Atomic)

Triggers when an IP datagram is received with the protocol field in the IP header set to 1 (ICMP), the Last Fragment bit is set, and

```
( IP offset * 8 ) + (IP data length)  > 65535
```

In other words, the IP offset (which represents the starting position of this fragment in the original packet, and which is in 8-byte units) plus the rest of the packet is greater than the maximum size for an IP packet.

3040 TCP - no bits set in flags (Attack, Atomic)

Triggers when a TCP packet is received with no bits set in the flags field.

3041 TCP - SYN and FIN bits set (Attack, Atomic)

Triggers when a TCP packet is received with both the SYN and FIN bits set in the flag field.

3042 TCP - FIN bit with no ACK bit in flags (Attack, Atomic)

Triggers when a TCP packet is received with the FIN bit set but with no ACK bit set in the flags field.

3050 Half-open SYN Attack/SYN Flood (Attack, Compound)

Triggers when multiple TCP sessions have been improperly initiated on any of several well-known service ports. Detection of this signature is currently limited to FTP, Telnet, HTTP, and e-mail servers (TCP ports 21, 23, 80, and 25 respectively).

3100 Smail Attack (Attack, Compound)

Triggers on the very common “smail” attack against SMTP-compliant e-mail servers (frequently sendmail).

3101 Sendmail Invalid Recipient (Attack, Compound)

Triggers on any mail message with a “pipe” (|) symbol in the recipient field.

3102 Sendmail Invalid Sender (Attack, Compound)

Triggers on any mail message with a “pipe” (|) symbol in the “From:” field.

3103 Sendmail Reconnaissance (Attack, Compound)

Triggers when “expn” or “vrfy” commands are issued to the SMTP port.

3104 Archaic Sendmail Attacks (Attack, Compound)

Triggers when “wiz” or “debug” commands are issued to the SMTP port.

3105 Sendmail Decode Alias (Attack, Compound)

Triggers on any mail message with “: decode@” in the header.

3106 Mail Spam (Attack, Compound)

Counts number of Rcpt to: lines in a single mail message and alarms after a user-definable maximum has been exceeded (default is 250).

3107 Majordomo Execute Attack (Attack, Compound)

A bug in the Majordomo program will allow remote users to execute arbitrary commands at the privilege level of the server.

3150 FTP Remote Command Execution (Attack, Compound)

Triggers when someone tries to execute the FTP SITE command.

3151 FTP SYST Command Attempt (Info, Compound)

Triggers when someone tries to execute the FTP SYST command.

3152 FTP CWD ~root (Attack, Compound)

Triggers when someone tries to execute the CWD ~root command.

3153 FTP Improper Address Specified (Attack, Atomic*)

Triggers if a port command is issued with an address that is not the same as the requesting host.

3154 FTP Improper Port Specified (Attack, Atomic*)

Triggers if a port command is issued with a data port specified that is less than 1024 or greater than 65535.

4050 UDP Bomb (Attack, Atomic)

Triggers when the UDP length specified is less than the IP length specified.

4100 Tftp Passwd File (Attack, Compound)

Triggers on an attempt to access the passwd file (typically /etc/passwd) via TFTP.

6100 RPC Port Registration (Info, Atomic*)

Triggers when attempts are made to register new RPC services on a target host.

6101 RPC Port Unregistration (Info, Atomic*)

Triggers when attempts are made to unregister existing RPC services on a target host.

6102 RPC Dump (Info, Atomic*)

Triggers when an RPC dump request is issued to a target host.

6103 Proxied RPC Request (Attack, Atomic*)

Triggers when a proxied RPC request is sent to the portmapper of a target host.

6150 ypserv Portmap Request (Info, Atomic*)

Triggers when a request is made to the portmapper for the YP server daemon (ypserv) port.

6151 ypbind Portmap Request (Info, Atomic*)

Triggers when a request is made to the portmapper for the YP bind daemon (ypbind) port.

6152 yppasswdd Portmap Request (Info, Atomic*)

Triggers when a request is made to the portmapper for the YP password daemon (yppasswdd) port.

6153 ypupdated Portmap Request (Info, Atomic*)

Triggers when a request is made to the portmapper for the YP update daemon (ypupdated) port.

6154 ypxfrd Portmap Request (Info, Atomic*)

Triggers when a request is made to the portmapper for the YP transfer daemon (ypxfrd) port.

6155 mountd Portmap Request (Info, Atomic*)

Triggers when a request is made to the portmapper for the mount daemon (mountd) port.

6175 rexd Portmap Request (Info, Atomic*)

Triggers when a request is made to the portmapper for the remote execution daemon (rexd) port.

6180 rexd Attempt (Info, Atomic*)

Triggers when a call to the rexd program is made. The remote execution daemon is the server responsible for remote program execution. This may be indicative of an attempt to gain unauthorized access to system resources.

6190 statd Buffer Overflow (Attack, Atomic*)

Triggers when a large statd request is sent. This could be an attempt to overflow a buffer and gain access to system resources.

8000 FTP Retrieve Password File (Attack, Atomic*)

SubSig ID: 2101

Triggers on string “passwd” issued during an FTP session. May indicate someone attempting to retrieve the password file from a machine in order to crack it and gain unauthorized access to system resources.

Cisco IOS Firewall IDS Configuration Task List

See the following sections for configuration tasks for the Cisco IOS Firewall Intrusion Detection System feature. Each task in the list is identified as optional or required:

- Initializing Cisco IOS Firewall IDS (Required)
- Initializing the Post Office (Required)
- Configuring and Applying Audit Rules (Required)
- Verifying the Configuration (Optional)

For examples using the commands in this chapter, see the “Cisco IOS Firewall IDS Configuration Examples” section at the end of this chapter.

Initializing Cisco IOS Firewall IDS

To initialize Cisco IOS Firewall IDS on a router, use the following commands in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **ip audit smtp spam** *recipients*	Sets the threshold beyond which spamming in e-mail messages is suspected. Here, *recipients* is the maximum number of recipients in an e-mail message. The default is 250.
Step 2	Router(config)# **ip audit po max-events** *number_events*	Sets the threshold beyond which queued events are dropped from the queue for sending to the Cisco Secure IDS Director. Here, *number_events* is the number of events in the event queue. The default is 100. Increasing this number may have an impact on memory and performance, as each event in the event queue requires 32 KB of memory.
Step 3	Router(config)# **exit**	Exits global configuration mode.

Initializing the Post Office

Note You must reload the router every time you make a Post Office configuration change.

To initialize the Post Office system, use the following commands in global configuration mode:

	Command	Purpose
Step 1	Router(config)# **ip audit notify nr-director** or Router(config)#**ip audit notify log**	Sends event notifications (alarms) to either a Cisco Secure IDS Director, a syslog server, or both. For example, if you are sending alarms to a Cisco Secure IDS Director, use the **nr-director** keyword in the command syntax. If you are sending alarms to a syslog server, use the **log** keyword in the command syntax.
Step 2	router(config)# **ip audit po local hostid** *host-id* **orgid** *org-id*	Sets the Post Office parameters for both the router (using the **ip audit po local** command) and the Cisco Secure IDS Director (using the **ip audit po remote** command). Here, *host-id* is a unique number between 1 and 65535 that identifies the router, and *org-id* is a unique number between 1 and 65535 that identifies the organization to which the router and Director both belong.

	Command	Purpose
Step 3	Router(config)# **ip audit po remote hostid** *host-id* **orgid** *org-id* **rmtaddress** *ip-address* **localaddress** *ip-address* **port** *port-number* **preference** *preference-number* **timeout** *seconds* **application** *application-type*	Sets the Post Office parameters for both the Cisco Secure IDS Director (using the **ip audit po remote** command). • *host-id* is a unique number between 1 and 65535 that identifies the Director. • *org-id* is a unique number between 1 and 65535 that identifies the organization to which the router and Director both belong. • **rmtaddress** *ip-address* is the Director's IP address. • **localaddress** *ip-address* is the router's interface IP address. • *port-number* identifies the UDP port on which the Director is listening for alarms (45000 is the default). • *preference-number* is the relative priority of the route to the Director (1 is the default)—if more than one route is used to reach the same Director, then one must be a primary route (preference 1) and the other a secondary route (preference 2). • *seconds* is the number of seconds the Post Office waits before it determines that a connection has timed out (5 is the default). • *application-type* is either **director** or **logger**. **Note** If you are sending Post Office notifications to a Sensor, use **logger** instead of **director** as your application. Sending to a logging application means that no alarms are sent to a GUI; instead, the Cisco Secure IDS alarm data is written to a flat file, which can then be processed with filters, such as **perl** and **awk**, or staged to a database. Use **logger** only in advanced applications where you want the alarms only to be logged and not displayed.
Step 4	Router(config)# **logging console info**	Displays the syslog messages on the router console if you are sending alarms to the syslog console.
Step 5	Router(config)# **exit**	Exits global configuration mode.
Step 6	Router# **write memory**	Saves the configuration.
Step 7	Router# **reload**	Reloads the router.

After you have configured the router, add the Cisco IOS Firewall IDS router's Post Office information to the */usr/nr/etc/hosts* and */usr/nr/etc/routes* files on the Cisco Secure IDS Sensors and Directors communicating with the router. You can do this with the nrConfigure tool in Cisco Secure IDS. For more information, refer to the *NetRanger User Guide*.

Configuring and Applying Audit Rules

To configure and apply audit rules, use the following commands starting in global configuration mode:

	Command	Purpose

Step 1	`Router(config)#` **`ip audit info`** `{`**`action`** `[`**`alarm`**`] [`**`drop`**`] [`**`reset`**`]}`
	and
	`Router(config)#` **`ip audit attack`** `{`**`action`** `[`**`alarm`**`] [`**`drop`**`] [`**`reset`**`]}`
		Sets the default actions for info and attack signatures. Both types of signatures can take any or all of the following actions: alarm, drop, and reset. The default action is **alarm**.

Step 2	`Router(config)#` **`ip audit name`** `audit-name {`**`info`** `|` **`attack`**`} [`**`list`** `standard-acl] [`**`action`** `[`**`alarm`**`] [`**`drop`**`] [`**`reset`**`]]`	Creates audit rules, where *audit-name* is a user-defined name for an audit rule. For example:
		`ip audit name` *`audit-name`* **`info`**
		`ip audit name` *`audit-name`* **`attack`**
		The default action is **alarm**.
		Note Use the same name when you assign attack and info type signatures.
		You can also use the **ip audit name** command to attach access control lists to an audit rule for filtering out sources of false alarms. In this case *standard-acl* is an integer representing an ACL. If you attach an ACL to an audit rule, the ACL must be defined as well:
		`ip audit name` *`audit-name`* `{`**`info`**`|`**`attack`**`}` **`list`** *`acl-list`*
		In the following example, ACL 99 is attached to the audit rule INFO, and ACL 99 is defined:
		`ip audit name INFO info list 99`
		`access-list 99 deny 10.1.1.0 0.0.0.255`
		`access-list 99 permit any`
		Note The ACL in the preceding example is *not* denying traffic from the 10.1.1.0 network (as expected if it were applied to an interface). Instead, the hosts on that network are not filtered through the audit process because they are trusted hosts. On the other hand, all other hosts, as defined by **permit any**, are processed by the audit rule.

	Command	Purpose

Step 3	Router(config)# **ip audit signature** signature-id {**disable** | **list** acl-list}	Disables individual signatures. Disabled signatures are not included in audit rules, as this is a global configuration change:
		ip audit signature *signature-number* **disable**
		To re-enable a disabled signature, use the **no ip audit signature** command, where *signature-number* is the number of the disabled signature.
		You can also use the **ip audit signature** command to apply ACLs to individual signatures for filtering out sources of false alarms. In this case *signature-number* is the number of a signature, and *acl-list* is an integer representing an ACL:
		ip audit signature *signature-number* **list** *acl-list*
		For example, ACL 35 is attached to the 1234 signature, and then defined:
		`ip audit signature 1234 list 35`
		`access-list 35 deny 10.1.1.0 0.0.0.255`
		`access-list 35 permit any`
		Note The ACL in the preceding example is *not* denying traffic from the 10.1.1.0 network (as expected if it were applied to an interface). Instead, the hosts on that network are not filtered through the signature because they are trusted hosts or are otherwise causing false positives to occur. On the other hand, all other hosts, as defined by **permit any**, are processed by the signature.

Step 4	Router(config-if)# **interface interface-number**	Enters interface configuration mode.
Step 5	Router(config-if)# **ip audit** *audit-name* {**in** | **out**}	Applies an audit rule at an interface. With this command, *audit-name* is the name of an existing audit rule, and *direction* is either **in** or **out**.
Step 6	Router(config-if)# **exit**	Exits interface configuration mode.
Step 7	Router(config)# **ip audit po protected** ip-addr [**to** ip-addr]	Configures which network should be protected by the router. Here, *ip_addr* is the IP address to protect.
Step 8	Router(config)# **exit**	Exits global configuration mode.

Verifying the Configuration

You can verify that Cisco IOS Firewall IDS is properly configured with the **show ip audit configuration** command (see Example 1).

Example 1 Output from show ip audit configuration Command

```
ids2611# show ip audit configuration

Event notification through syslog is enabled
Event notification through Net Director is enabled
Default action(s) for info signatures is alarm
Default action(s) for attack signatures is alarm drop reset
Default threshold of recipients for spam signature is 25
PostOffice:HostID:55 OrgID:123 Msg dropped:0
          :Curr Event Buf Size:100  Configured:100
HID:14 OID:123 S:1 A:2 H:82 HA:49 DA:0 R:0 Q:0
 ID:1 Dest:10.1.1.99:45000 Loc:172.16.58.99:45000 T:5 S:ESTAB *

Audit Rule Configuration
 Audit name AUDIT.1
  info actions alarm
  attack actions alarm drop reset
```

You can verify which interfaces have audit rules applied to them with the **show ip audit interface** command (see Example 2).

Example 2 Output from show ip audit interface Command

```
ids2611# show ip audit interface

Interface Configuration
 Interface Ethernet0
 Inbound IDS audit rule is AUDIT.1
  info actions alarm
  attack actions alarm drop reset
 Outgoing IDS audit rule is not set
Interface Ethernet1
 Inbound IDS audit rule is AUDIT.1
  info actions alarm
  attack actions alarm drop reset
 Outgoing IDS audit rule is not set
```

Monitoring and Maintaining Cisco IOS Firewall IDS

This section describes the EXEC commands used to monitor and maintain Cisco IOS Firewall IDS.

Command	Purpose
Router# **clear ip audit configuration**	Disables Cisco IOS Firewall IDS, removes all intrusion detection configuration entries, and releases dynamic resources.
Router# **clear ip audit statistics**	Resets statistics on packets analyzed and alarms sent.
Router# **show ip audit statistics**	Displays the number of packets audited and the number of alarms sent, among other information.

The following display provides sample output from the **show ip audit statistics** command:

```
Signature audit statistics [process switch:fast switch]
 signature 2000 packets audited: [0:2]
 signature 2001 packets audited: [9:9]
 signature 2004 packets audited: [0:2]
 signature 3151 packets audited: [0:12]
Interfaces configured for audit 2
Session creations since subsystem startup or last reset 11
Current session counts (estab/half-open/terminating) [0:0:0]
Maxever session counts (estab/half-open/terminating) [2:1:0]
Last session created 19:18:27
Last statistic reset never

HID:1000 OID:100 S:218 A:3 H:14085 HA:7114 DA:0 R:0
```

Cisco IOS Firewall IDS Configuration Examples

The following sections provide Cisco IOS Firewall IDS configuration examples:

- Cisco IOS Firewall IDS Reporting to Two Directors Example
- Adding an ACL to the Audit Rule Example
- Disabling a Signature Example
- Adding an ACL to Signatures Example
- Dual-Tier Signature Response Example

Cisco IOS Firewall IDS Reporting to Two Directors Example

In the following example, Cisco IOS Firewall IDS is initialized. Notice that the router is reporting to two Directors. Also notice that the AUDIT.1 audit rule will apply both info and attack signatures.

```
ip audit smtp spam 25
ip audit notify nr-director
ip audit notify log
ip audit po local hostid 55 orgid 123
ip audit po remote hostid 14 orgid 123 rmtaddress 10.1.1.99 localaddress 10.1.1.1
ip audit po remote hostid 15 orgid 123 rmtaddress 10.1.2.99 localaddress 10.1.1.1

ip audit name AUDIT.1 info action alarm
ip audit name AUDIT.1 attack action alarm drop reset

interface e0
 ip address 10.1.1.1 255.0.0.0
 ip audit AUDIT.1 in

interface e1
 ip address 172.16.57.1 255.255.255.0
 ip audit AUDIT.1 in
```

Adding an ACL to the Audit Rule Example

In the following example, an ACL is added to account for a Cisco Secure IDS Scanner (172.16.59.16) that scans for all types of attacks. As a result, no packets originating from the device will be audited.

```
ip audit smtp spam 25
ip audit notify nr-director
ip audit notify log
ip audit po local hostid 55 orgid 123
ip audit po remote hostid 14 orgid 123 rmtaddress 10.1.1.99 localaddress 10.1.1.1
ip audit po remote hostid 15 orgid 123 rmtaddress 10.1.2.99 localaddress 10.1.1.1

ip audit name AUDIT.1 info list 90 action alarm
ip audit name AUDIT.1 attack list 90 action alarm drop reset

interface e0
 ip address 10.1.1.1 255.0.0.0
 ip audit AUDIT.1 in

interface e1
 ip address 172.16.57.1 255.255.255.0
 ip audit AUDIT.1 in

access-list 90 deny 172.16.59.16
access-list 90 permit any
```

Disabling a Signature Example

The security administrator notices that the router is generating a lot of false positives for signatures 1234, 2345, and 3456. The system administrator knows that there is an application on the network that is causing signature 1234 to fire, and it is not an application that should cause security concerns. This signature can be disabled, as illustrated in the following example:

```
ip audit smtp spam 25
ip audit notify nr-director
ip audit notify log
ip audit po local hostid 55 orgid 123
ip audit po remote hostid 14 orgid 123 rmtaddress 10.1.1.99 localaddress 10.1.1.1
ip audit po remote hostid 15 orgid 123 rmtaddress 10.1.2.99 localaddress 10.1.1.1


ip audit signature 1234 disable

ip audit name AUDIT.1 info list 90 action alarm
ip audit name AUDIT.1 attack list 90 action alarm drop reset

interface e0
 ip address 10.1.1.1 255.0.0.0
 ip audit AUDIT.1 in

interface e1
 ip address 172.16.57.1 255.255.255.0
 ip audit AUDIT.1 in

access-list 90 deny 172.16.59.16
access-list 90 permit any
```

Adding an ACL to Signatures Example

After further investigation, the security administrator discovers that the false positives for signatures 2345 and 3456 are caused by specific applications on hosts 10.4.1.1 and 10.4.1.2, as well as by some workstations using DHCP on the 172.16.58.0 subnetwork. Attaching an ACL that denies processing of these hosts stops the creation of false positive alarms, as illustrated in the following example:

```
ip audit smtp spam 25
ip audit notify nr-director
ip audit notify log
ip audit po local hostid 55 orgid 123
ip audit po remote hostid 14 orgid 123 rmtaddress 10.1.1.99 localaddress 10.1.1.1
ip audit po remote hostid 15 orgid 123 rmtaddress 10.1.2.99 localaddress 10.1.1.1

ip audit signature 1234 disable
ip audit signature 2345 list 91
ip audit signature 3456 list 91

ip audit name AUDIT.1 info list 90 action alarm
ip audit name AUDIT.1 attack list 90 action alarm drop reset

 interface e0
ip address 10.1.1.1 255.0.0.0
 ip audit AUDIT.1 in

interface e1
 ip address 172.16.57.1 255.255.255.0
 ip audit AUDIT.1 in

access-list 90 deny 172.16.59.16
access-list 90 permit any
access-list 91 deny host 10.4.1.1
access-list 91 deny host 10.4.1.2
access-list 91 deny 172.16.58.0 0.0.0.255
access-list 91 permit any
```

Dual-Tier Signature Response Example

The company has now reorganized and has placed only trusted people on the 172.16.57.0 network. The work done by the employees on these networks must not be disrupted by Cisco IOS Firewall IDS, so attack signatures in the AUDIT.1 audit rule now will only alarm on a match.

For sessions that originate from the outside network, any attack signature matches (other than the false positive ones that are being filtered out) are to be dealt with in the following manner: send an alarm, drop the packet, and reset the TCP session.

This dual-tier method of signature response is accomplished by configuring two different audit specifications and applying each to a different ethernet interface, as illustrated in the following example:

```
ip audit smtp spam 25
ip audit notify nr-director
ip audit notify log
ip audit po local hostid 55 orgid 123
ip audit po remote hostid 14 orgid 123 rmtaddress 10.1.1.99 localaddress 10.1.1.1
ip audit po remote hostid 15 orgid 123 rmtaddress 10.1.2.99 localaddress 10.1.1.1

ip audit signature 1234 disable
ip audit signature 2345 list 91
ip audit signature 3456 list 91
ip audit name AUDIT.1 info list 90 action alarm
ip audit name AUDIT.1 attack list 90 action alarm
ip audit name AUDIT.2 info action alarm
ip audit name AUDIT.2 attack alarm drop reset

interface e0
    ip address 10.1.1.1 255.0.0.0
    ip audit AUDIT.2 in

interface e1
    ip address 172.16.57.1 255.255.255.0
    ip audit AUDIT.1 in

access-list 90 deny host 172.16.59.16
access-list 90 permit any
access-list 91 deny host 10.4.1.1
access-list 91 deny host 10.4.1.2
access-list 91 deny 172.16.58.0 0.0.0.255
access-list 91 permit any
```